

SpaceShooter2D Online 版实作范例

- 1-1 Online 基本概念
- 1-2 arcalet 入门基础
- 1-3 登入游戏
- 1-4 处理讯息
- 1-5 玩家操控

八风吹不动，一屁打过江

《佛印禅师公案》

射击游戏是实时要求非常高的游戏，为了让程序设计师很快能学会设计在线游戏程序，所以我们选择这个简单的游戏做为范例。

依续着上一章的单机版太空射击游戏，在改造为多人联机后，比较单机与在线的版本，您会发现用 arcalet 设计在线游戏真的很简单，原本如如不动的远程场景对象，在一弹指间竟然随着远程玩家的动作同步动了起来。

1-1 Online 基本概念

1-2 arcalet 程序设计基础

有网络程序经验的开发者可能会翻查 **arcalet** API 参考手册, 想要找到跟网络有关的 API 对象类别。幸运的是, 使用 **arcalet** 服务并不需要撰写网络程序, 当然 API 手册里也不会有网络相关的对象类别。

开发在线游戏不需要撰写网络通讯程序, **arcalet** 颠覆了在线游戏程序设计思维。本文尝试将上一章所制作的太空射击游戏改造为可以多人联机, 让每个玩家都在同一个太空场景中操控属于自己的太空战机, 共同迎战陨石的攻击的「SpaceShooter2D Online」。

API 对象类别

arcalet API 总共有四个基本服务的对象类别, 分别简单说明如下:

■ ArcaletGame

建立 **ArcaletGame** 对象将会让客户端程序与 **arcalet** 游戏服务器建立联机并登入系统, 在联机的状态的下才能与其他玩家实时互动, 也才能使用 **ArcaletScene**、**ArcaletItem**、**ArcaletScore** 等对象。

一般来说, 玩家登入网络游戏后应该马上进入游戏大厅, 因此 **arcalet** 服务让每一个游戏都会有一个默认的静态场景做为大厅场景。大厅场景跟一般游戏场景不同, 玩家不需要进入, 也不能离开, 大厅场景在玩家登入服务器后已经存在, 并且自动进入场景, 除非联机终止, 否则玩家会持续收到大厅中的由其它玩家传来的讯息, 也随时可以发送讯息至大厅。

游戏进行有时也会传递私人讯息, 因此游戏对象里面也有一个特殊的场景, 我们称之为私人场景。私人场景跟大厅场景一样不需要进入, 也不能离开, 它在游戏联机之后就已经存在, 直到联机终止。

■ ArcaletScene

场景是游戏玩家互动的基础, 在 **arcalet** 的场景观念里, 玩家可以同时身处多个场景, 其规划与应用方式由游戏开发者自行决定。

场景又分为「静态场景」与「动态场景」, 「静态场景」不会有多个副本(instance), 只能进入、离开, 不能创建; 「动态场景」则可依开发者与玩家的需要建立新的副本, 因此可以被创建而产生一个副本, 已经创建的动态场景副本才能进入与离开。

■ ArcaletItem

物品又称虚拟宝物，此对象用来管理玩家在各种游戏中的各种物品。不过 ArcaletItem 可不是只能管理虚拟宝物这么简单，它还可以用来做为记录玩家或游戏状态的永久性内存。

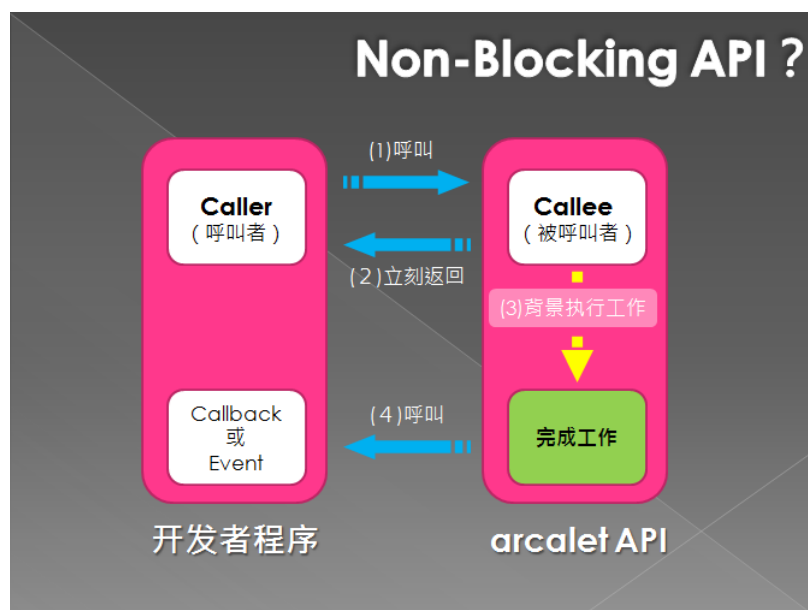
■ ArcaletScore

记录游戏分数，由服务器统计分数，并产生计分排行榜(Leaderboard)。

异步的 Non-Blocking API

为了避免游戏画面冻结，许多 arcalet API 时都被设计成异步的 Non-Blocking 模式，因此呼叫时必须指定 Callback 函式，或是在呼叫前指定 Event 处理函式，好让 API 完成工作后透过 Callback 或 Event 函式通知呼叫者，至于何时使用 Callback，何时使用 Event，则交由 API 函式的定义来决定。

Callback 和 Event 是开发者自己所设计的函式，由在背景中执行的 API 工作程序所呼叫，如果 API 完成工作后有数据要传回，就要把这些数据当成参数，在呼叫 callback 函式或 Event 处理函式时传回给开发者程序。



第一步: 建立 arcalet 主控场景

使用 arcalet 开发 Unity3D 游戏必须先建立一个 arcalet 主控场景，并且要将这个场景设定为游戏开始后的第一个场景。主控场景内只有一个空的 GameObject，内有 arcalet 主控程序，这个主控程序也就是游戏项目的通讯系统核心。

Note

「arcalet 主控场景」是 Unity3D 的场景，而非游戏服务器中的场景。

为了方便说明，我们定义了几个名词，下表说明了这些名词的意义：

■ AGCCMain

主控场景名称

■ AX

arcalet 主控器，同时也是一个 GameObject

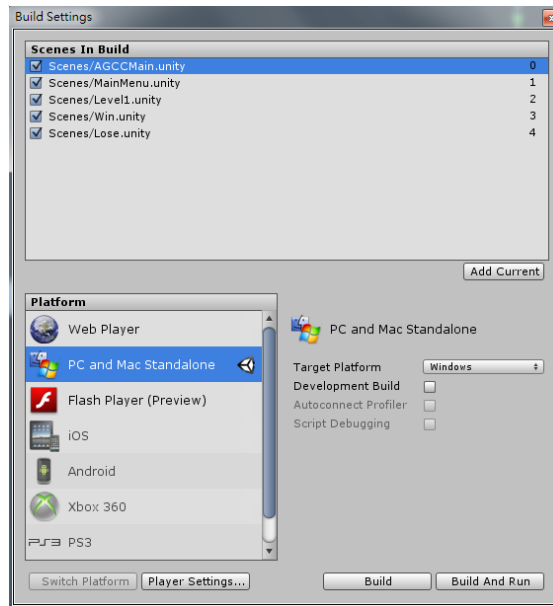
■ AGCC

arcalet 主控器 AX 的控制脚本

以下是建立 arcalet 主控场景的详细步骤，请确实按照这些步骤建立 AGCCMain：

- 一、将 arcalet.dll 放置到项目的 Assets 中，这个 DLL 档请至 arcalet 开发者网站 <http://developer.arcalet.com> 下载
- 二、建立一个空的 Scene，取名为 AGCCMain，随后存档为 AGCCMain.unity。
- 三、在「GameObject」的「Create Empty」建立一个空的 Game Object，更名为 AX。
- 四、在「Project」窗中新增一个 C# 脚本，更名为 AGCC，然后拖拉至 AX 中。
- 五、把 AGCCMain 设定为第一个进入的 Scene：

从「Build Settings」中的「Add Current」AGCCMain 加入「Build In Scene」，然后使用鼠标拖曳把 AGCCMain 拉到最上面，成为此项目的第一个 Scene。



建立 AGCC 脚本的样版程序代码

以下是 AGCC 脚本的样版程序代码，请直接复制贴到 AGCC.cs 程序文件，请注意，此处有三个变量 gguid、sguid、gcert 并没有指定正确的初始值，稍后会告诉您如何取得正确的值。

```
// AGCC.cs
using UnityEngine;
using System.Collections;

public class AGCC : MonoBehaviour {

    public ArcaletGame ag=null;

    private const string gguid="请将游戏的GGUID 放在这里";
    private const string sguid="请将游戏主大厅的SGUID 放在这里";
    private byte[] gcert = {请将游戏的认证码GCERT 放在这里};

    private string userid;
    private bool ArcaletGameHasLaunched=false;

    void Awake() {
        DontDestroyOnLoad(this);
        ArcaletGameHasLaunched=false;
    }

    // 马上进入第一个场景，也就是单机版的主场景
    void Start () {
        Application.LoadLevel("MainMenu");
    }

    void Update () {
        if (ArcaletGameHasLaunched) ag.EventDispatcher();
    }

    // 按下UNITY editor上的「停止」钮
    void OnApplicationQuit() {
        if (ArcaletGameHasLaunched) ag.Dispose();
    }
}
```

```
public void ArcaletStartup(string userid,string passwd) {  
    ag=new ArcaletGame(userid,passwd,gguid,sguid,gcert);  
    this.userid=userid;  
    ag.STALaunch();  
    ArcaletGameHasLaunched=true;  
}  
}
```

此处我们设计了一个函式 `ArcaletStartup()`，用来进行网络联机与登入游戏服务器，它透过 `ArcaletGame` 建构函式产生对象，并存放于变量 `ag`，真正进行联机与登入则是呼叫 `ag.Launch()`，由于 **arcalet** API 采用 `non-blocking` 方式进行背景处理，所以 `Launch()` 函式会马上返回，所以 `ArcaletStartup()` 函式将来也可以在 `Update()`、`OnGUI()`、`FixedUpdate()` 等函式中被呼叫，不必担心造成 `frame` 延迟。

取得建立 `ArcaletGame` 的系统参数

`GGUID`、`SGUID`、`GCERT`(Game Certificate)是建立 `ArcaletGame` 对象的三个重要参数，开发者只要到 **arcalet** 开发者后台建立一个新的游戏便可取得这三个参数，步骤如下：

- 一、开发者可至 **arcalet** 开发者网站 (<http://developer.arcalet.com>) 免费申请成为 **arcalet** 的开发者，完成信箱认证后便可开始制作在线游戏，也可以获得在线技术支持与开发者交流等服务。



arcalet 开发者网站首页及开发者申请表单

Note

arcalet 网站常会不定期更改，除了增加新的功能外，画面也可能本文的截图不太一样。

二、成功获得开发者身份后便可登入到 **arcalet** 开发者管理后台，然后点选「新增游戏名称」，为游戏项目建立一个专属的服务项目。下图是登入开发者后台的第一个画面，因为是初次登入，所以没有任何属于这个开发者的游戏。

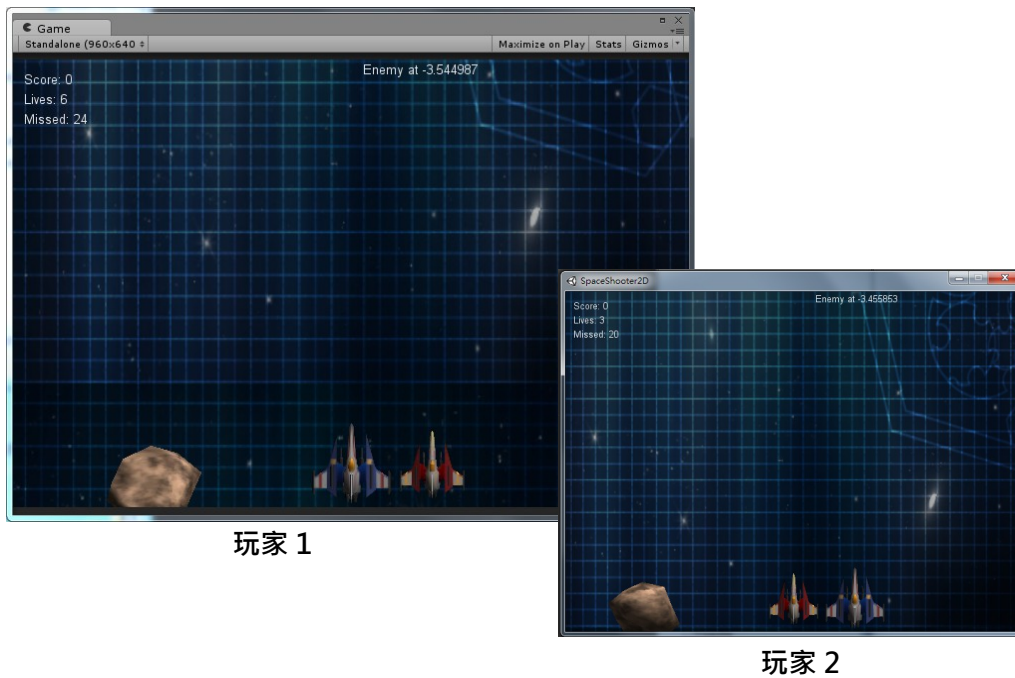


三、每一項游戏专属空间建立好后，**arcalet** 就会为该项目产生这重要的三个参数(GGUID、SGUID、Game Certificate):



GGUID、SGUID、Game Certificate 等三个参数是玩家登入游戏所必需的，稍后的内容将会介绍如何使用这些参数与游戏服务器建立联机。另外，我们 **arcalet** 开发者后台看到了许多和在线游戏开发与营运有关的各项功能，由于本文着重在多人联机互动，所以只会用到其中的场景功能，其他详细的服务内容将在未来另辟专章介绍。

从无到有—实作运行结果



完整专案下载请至：

<http://developer.arcalet.com>